What is claimed is:

1. A method of imputing missing values in microarray data comprising the steps of:

(a) clustering the data by a Gaussian mixture clustering model; and

(b) estimating missing values by a GMCimpute algorithm

thereby imputing missing values in microarray data.

2. The method of claim 1, wherein the Gaussian mixture clustering model comprises the steps of

(a) determining a value of *K*;

(b) partitioning the rows of the microarray data into *K* partitions; and

(c) repeating a Classification Expectation-Maximization algorithm until the *K* partitions converge.

3. A computer program product comprising a computer software program, wherein the computer software program, once executed by a computer processor, performs a method of imputing missing values in microarray data according to the method of claim 1.

4. The computer program product of claim 3, wherein the Gaussian mixture clustering model comprises the steps of

(a) determining a value of *K*;

(b) partitioning the rows of the microarray data into *K* partitions; and

(c) repeating a Classification Expectation-Maximization algorithm until the *K* partitions converge.

5. A computer software program, wherein the computer software program, once executed by a computer processor, performs a method of imputing missing values in microarray data according to the method of claim 1.

6. The computer software program of claim 5, wherein the Gaussian mixture clustering model comprises the steps of

(a) determining a value of *K*;

(b) partitioning the rows of the microarray data into *K* partitions; and

(c) repeating a Classification Expectation-Maximization algorithm until the *K* partitions converge.

7. A computer comprising a computer memory having a computer software program stored therein, wherein the computer software program, once executed by a computer processor, performs a method of imputing missing values in microarray data according to the method of claim 1.

8. The computer of claim 7 wherein the Gaussian mixture clustering model comprises steps of

(a) determining a value of *K*;

(b) partitioning the rows of the microarray data into *K* partitions; and

(c) repeating a Classification Expectation-Maximization algorithm until the *K* partitions converge.

```
EM_ESTIMATE (μ1, ..., μK, Σ1, ..., ΣK, τ1, ..., τK, A')
{           FOR EACH ROW R OF A' WITH MISSING VALUES
{           FOR i = 1, ..., K
{           USE EM AND N(μi, Σi) TO ESTIMATE THE
          MISSING VALUES IN R.
Ri ← R WITH MISSING VALUES REPLACED BY ESTIMATES.
                              }
            }
R' ← WEIGHTEDAVERAGE(R1, ..., RK).
REPLACE R IN A' BY R' .
RETURN A' .
}

K_ESTIMATE(K, A)
{           /* FIRST PART: INITIALIZATION */
B ← ROWS OF A WITHOUT MISSING VALUES.
μ1, ..., μK, Σ1, ..., ΣK, τ1, ..., τK ←
   GAUSSIAN MIXTURE CLUSTERING OF B.
A' ← EM_ESTIMATE (μ1, ..., μK, Σ1, ..., ΣK, τ1, ..., τK, A).
/* SECOND PART: ITERATION */
REPEAT
{           μ1, ..., μK, Σ1, ..., ΣK, τ1, ..., τK ←
   GAUSSIAN MIXTURE CLUSTERING OF A'.
A' ← EM_ESTIMATE (μ1, ..., μK, Σ1, ..., ΣK, τ1, ..., τK, A').
} UNTIL CONVERGENCE
}

GMCimpute(S, A)
{           FOR K = 1, 2, ..., S
{           AK ← K_ESTIMATE(K, A).
}
RETURN (A1+ A2 + ... + AS) / S.
}
```

FIG. 1